SAND2012-7388P

DAKOTA 101: Wrap-Up

http://dakota.sandia.gov

DAKOTA 101: Wrap-Up Topics

- Review of DAKOTA's scope and relevance
- Recommended best practices
- Sneak preview of advanced topics
  - Application interfacing
  - Parallelism
  - Hybrid and advanced algorithms
  - Parameter estimation
- Resources for getting started

DAKOTA Supports

DAKOTA includes a wide array of algorithm capabilities to support engineering transformation through advanced modeling and simulation.

- **Simulation-based engineering design: optimize virtual (computational) prototypes**
- **Risk analysis and quantification of margins and uncertainty (QMU): assess the effect of parametric uncertainty on the probability of achieving desired system performance**
- **Verification and validation: automate mesh convergence or solver tolerance studies, generate ensembles of possible simulations or statistics to compare to experimental data**

DAKOTA Explores Model Parameter Space to Answer Engineering Questions

disease kinetic parameters

epidemic size, duration, severity

Matlab ODE Epidemic Model

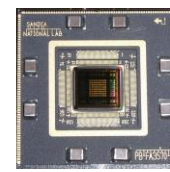
What are the crucial factors/parameters and how do they affect key metrics? *(sensitivity)*

What models and parameters best match experimental data? *(calibration)*

resistances, via diameters

voltage drop, peak current

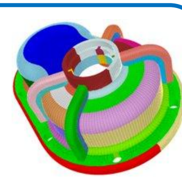
Xyce, Spice Circuit Model



material props, boundary, initial conditions

temperature, stress, flow rate

Abaqus, Sierra, CM/ CFD Model



How safe, reliable, robust, or variable is my system in the presence of uncertainties? *(UQ)*

What is the best performing design or control? *(optimization)*

load, modulus

stress, displacement

Cantilever Beam Model

Sandia National Laboratories

Basic Steps to Using DAKOTA

1. Define analysis goals; understand how DAKOTA helps and select a method to use
2. Access DAKOTA and understand help resources
3. **Workflow:** create an automated workflow so DAKOTA can communicate with your simulation (Advanced Topic)
   - Parameters to model, responses from model to DAKOTA
   - Typically requires scripting (Python, Perl, Shell, Matlab) or programming (C, C++, Java, Fortran)
   - Workflow usually crosscuts DAKOTA analysis types
4. **DAKOTA input file:** Jaguar GUI or text editor to configure DAKOTA to exercise the workflow to meet your goals
   - Tailor variables, methods, responses to analysis goals
5. Run DAKOTA: command-line; text input / output

Recommended Best Practices

- Test the building blocks
  - Test response extraction and interfaces before using with DAKOTA
  - Do a parameter study with a simple model
- Start with a parameter study
  - Screen problem characteristics: failure, smoothness, cost
  - Assess simulation robustness, verification, validity with respect to parameter variations
- Solicit expert help in
  - Formulating problem
  - Selecting appropriate methods
- Question the numbers
  - Sanity check aggregate/summary stats and results with more in-depth analyses of dakota-generated data.

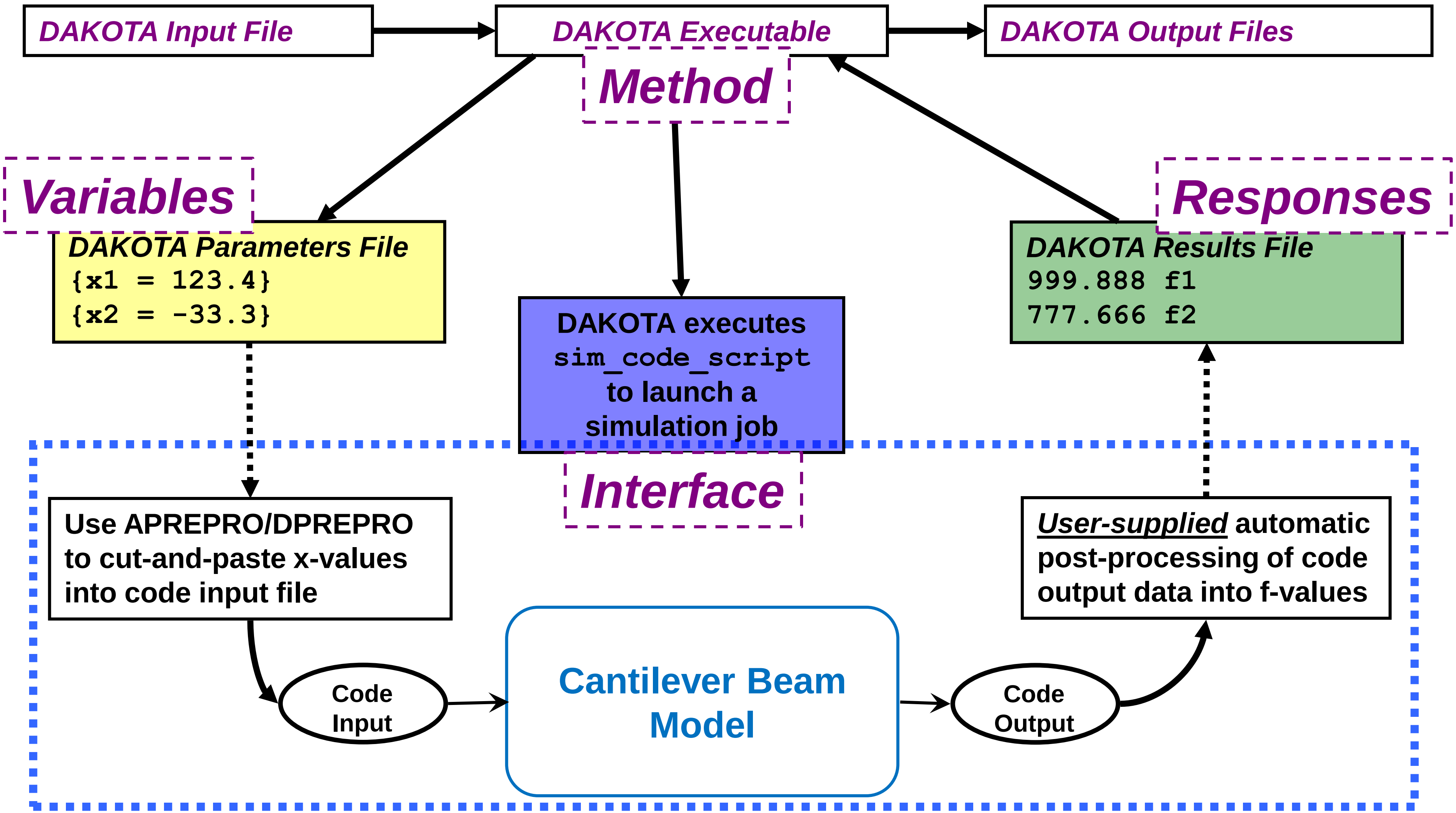
Discussion: DAKOTA Relevance Revisited

- Discuss your revised impressions of DAKOTA's relevance for your problems
- With what kinds of applications, simulations, computational models would you use it
- On what kinds of computer architecture would you want to use it (desktop workstation, Windows laptop, high-performance compute cluster)

Sneak Preview of Advanced Topics

- Application interfacing
  - Generic interface to simulations
  - Parameters in, responses out
- Parallelism
  - Computing on multi-core desktops, clusters, and capability platforms
  - Different levels of parallelism
- Hybrid and advanced algorithms
  - Time-tested and advanced algorithms
  - Strategies for combining methods
- Parameter Estimation
  - Additional problem formulations
  - Future capabilities
  - User requirements

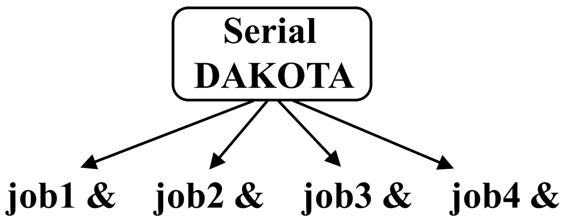
Interface communicates through file system and user-supplied script



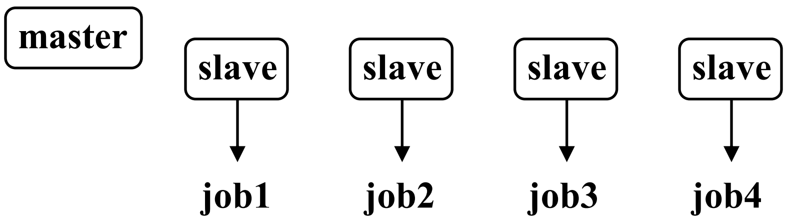
Parallelism from a computing platform perspective

Nested parallel models support large-scale applications and architectures.

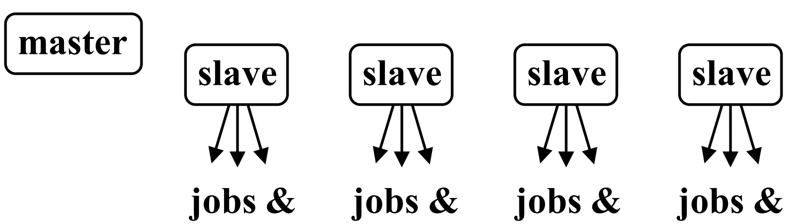
1. SMP/multiprocessor workstations: Asynchronous (external job allocation)



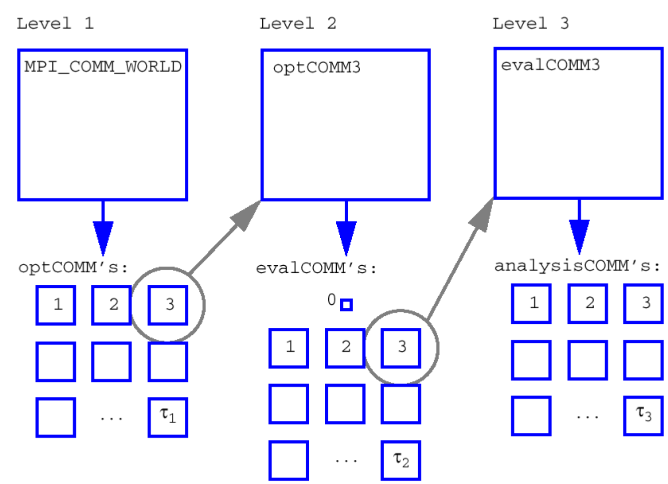
2. Cluster of workstations: Message-passing (internal job allocation)



3. Cluster of SMP's: Hybrid (service/compute model)



4. MPP: Internal MPI partitions (nested parallelism)



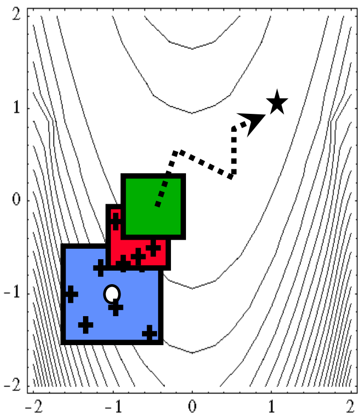
Sandia National Laboratories

Parallelism from an algorithmic perspective

1. ***Algorithmic coarse-grained parallelism***: **independent fn. Evaluations performed concurrently:**
   - **Gradient-based (e.g., finite difference gradients, speculative opt.)**
   - **Nongradient-based (e.g., GAs, PS, Monte Carlo)**
   - **Approximate methods (e.g., DACE)**
   - **Concurrent-method strategies (e.g., parallel B&B, island-model GAs, OUU)**
2. ***Algorithmic fine-grained parallelism:*** **computing the internal linear algebra of an opt. algorithm in parallel (e.g., large-scale opt., SAND)**
3. ***Function evaluation coarse-grained parallelism:*** **concurrent execution of separable simulations within a fn. eval. (e.g., multiple loading cases)**
4. ***Function evaluation fine-grained parallelism:*** **parallelization of the solution steps within a single analysis code (e.g., SALINAS, MPSalsa)**

Sandia National Laboratories

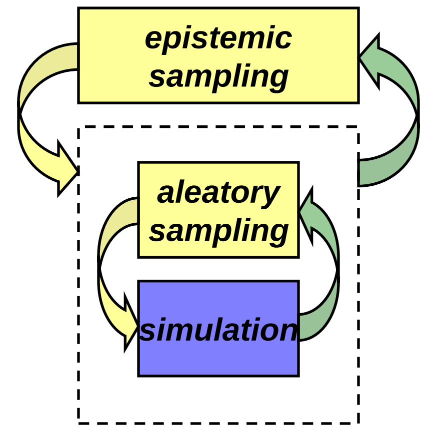
Opportunities for Mixing and Matching Methods



Strategies ***(general nesting, layering, sequencing and recasting facilities) combine methods to enable advanced studies:***

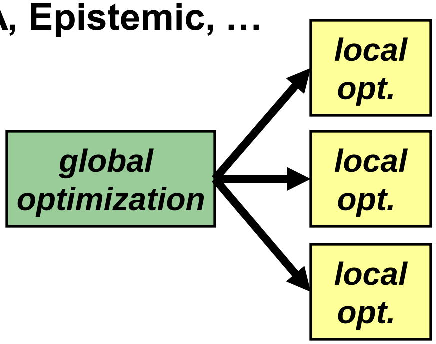
- **opt within opt (multilevel opt & hierarchical MDO)**
- **UQ within UQ (second-order probability)**
- **UQ within opt (OUU) and NLS (MCUU)**
- **opt within UQ (uncertainty of optima)**

with and without surrogate model indirection



Optimization

- **Surrogate-based: data fit, multifidelity, ROM**
- **Mixed integer nonlinear programming (MINLP): PEBBL (parallel branch and bound)**
- **Optimization under uncertainty**
  - **TR-SBOUU, RBDO (Bi-level, Sequential)**
  - **MCUU, PC-BDO, EGO/EGRA, Epistemic, …**
- **Hybrids (e.g., global/local)**
- **Pareto set**
- **Multi-start**
- **Multilevel methods**



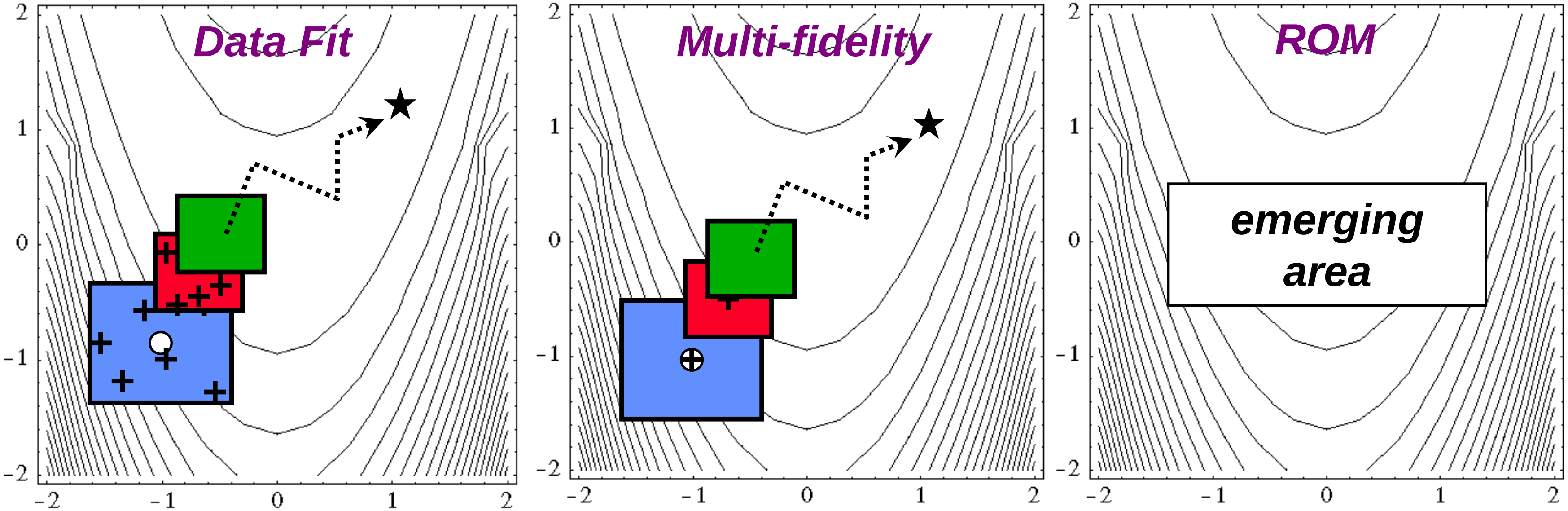
Uncertainty

- **Second order probability**
- **Uncertainty of optima**

Nonlinear least squares

- **Surrogate-based calibration**
- **Model calibration under uncertainty**

Trust Region Surrogate-Based Minimization



Data fit surrogates

- Global: polynomials, splines, neural network, Kriging, RBFs
- Local: 1st/2nd-order Taylor

Data fits in SBO

- Smoothing: extract global trend
- DACE: limited # design vars
- Must balance local consistency with global accuracy

Multifidelity surrogates:

- Coarser discretizations, looser conv. tols., reduced element order
- Omitted physics: e.g., Euler CFD, panel methods

Multifidelity SBO

- HF scale better w/ des. vars.
- Requires smooth LF model
- May require design mapping
- Correction quality is crucial

ROM surrogates:

- Spectral decomposition
- POD/PCA w/ SVD
- KL/PCE (random fields, stochastic processes)

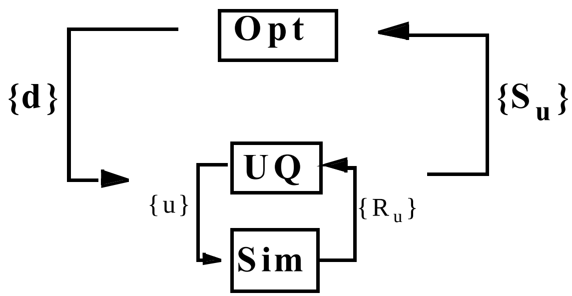
ROMs in SBO

- Key issue: parametrize (extended or spanning ROM)
- Otherwise like data fit case

Sandia National Laboratories

Optimization Under Uncertainty

Rather than design and then post-process to evaluate uncertainty…
actively design optimize while accounting for uncertainty/reliability metrics $s_u(d)$, e.g., mean, variance, reliability, probability:



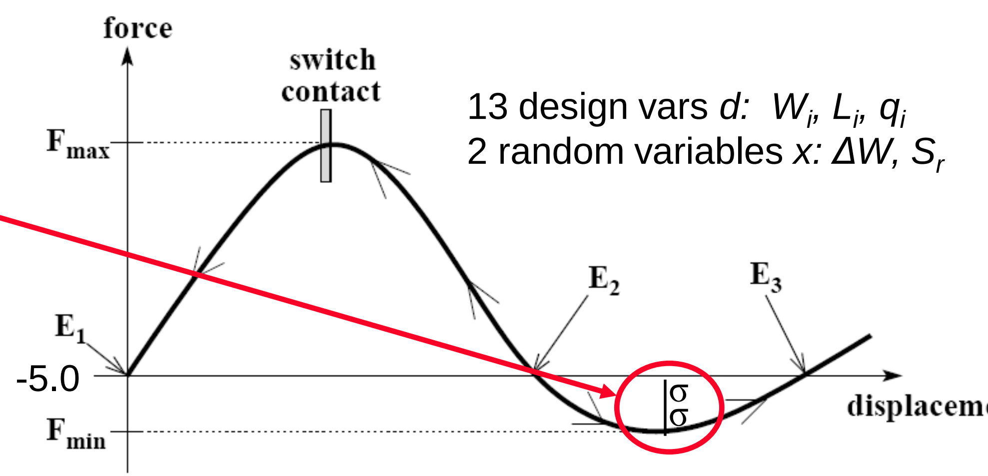
(nested paradigm)

$$
\begin{aligned}
\min \quad & f(d) + W\, s_u(d) \\
\text{s.t.} \quad & g_l \le g(d) \le g_u \\
& h(d) = h_t \\
& d_l \le d \le d_u \\
& a_l \le A_i\, s_u(d) \le a_u \\
& A_e\, s_u(d) = a_t
\end{aligned}
$$

Bistable switch problem formulation (Reliability-Based Design Optimization):

simultaneously reliable and robust designs

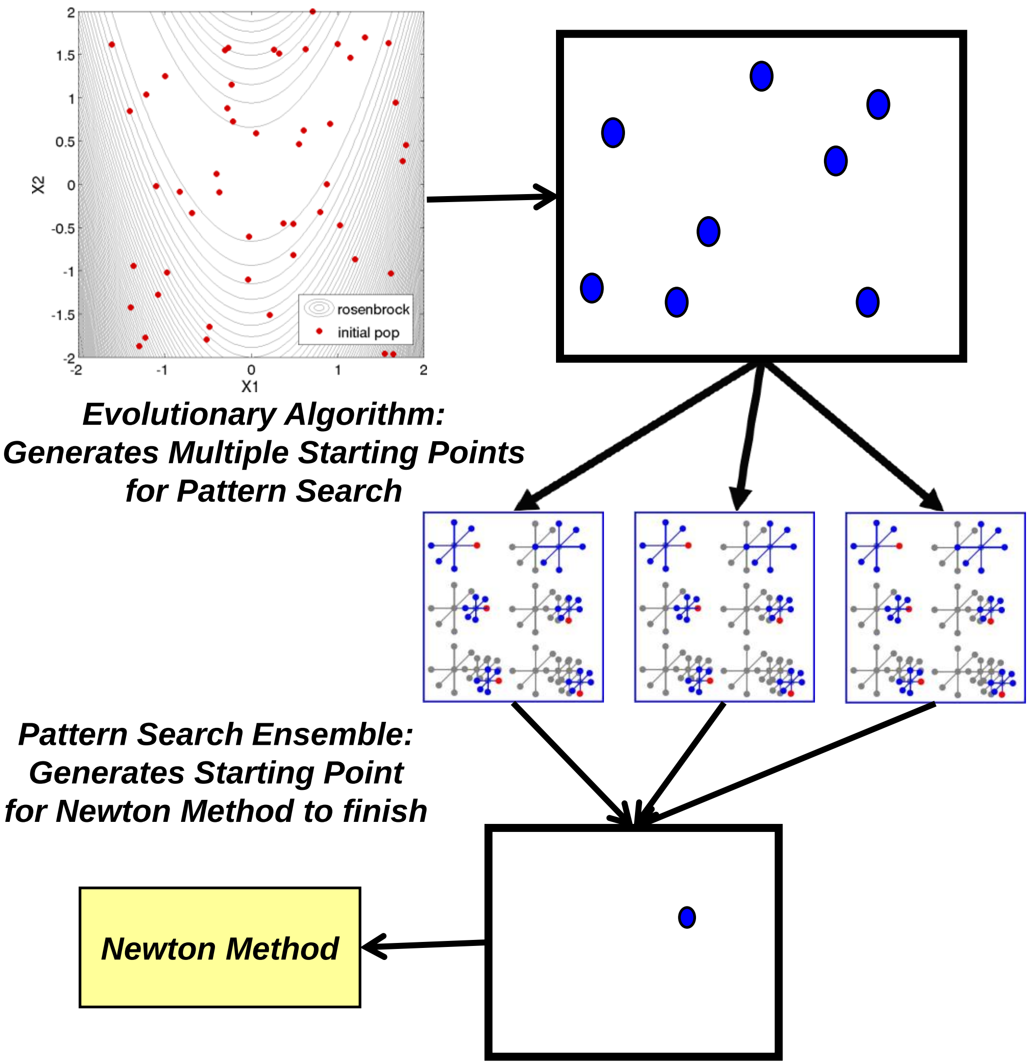
$$
\begin{array}{llcll}
\max & & & \mathrm{E}\left[F_{min}(\mathbf{d},\mathbf{x})\right] & \\
\text{s.t.} & 2 & \le & \beta_{ccdf}(\mathbf{d}) & \\
& 50 & \le & \mathrm{E}\left[F_{max}(\mathbf{d},\mathbf{x})\right] & \le 150 \\
& & & \mathrm{E}\left[E_2(\mathbf{d},\mathbf{x})\right] & \le 8 \\
& & & \mathrm{E}\left[S_{max}(\mathbf{d},\mathbf{x})\right] & \le 3000
\end{array}
$$



13 design vars *d:* W_i, L_i, q_i
2 random variables *x:* ΔW, S_r

Hybrid Optimization

```
strategy,
  graphics
  hybrid sequential
    method_list = 'GA' 'PS' 'NLP'
method,
  id_method = 'GA'
  model_pointer = 'M1'
  coliny_ea
    seed = 1234
    population_size = 10
    verbose output
method,
  id_method = 'PS'
  model_pointer = 'M1'
  coliny_pattern_search stochastic
    seed = 1234
    initial_delta = 0.1
    threshold_delta = 1.e-4
    solution_accuracy = 1.e-10
    exploratory_moves basic_pattern
    verbose output
method,
  id_method = 'NLP'
  model_pointer = 'M2'
  optpp_newton
    gradient_tolerance = 1.e-12
    convergence_tolerance = 1.e-15
    verbose output
model,
  id_model = 'M1'
  single
    variables_pointer = 'V1'
    interface_pointer = 'I1'
    responses_pointer = 'R1'
model,
  id_model = 'M2'
  single
    variables_pointer = 'V1'
    interface_pointer = 'I1'
    responses_pointer = 'R2'
variables,
  id_variables = 'V1'
  continuous_design = 2
    initial_point     0.6     0.7
    upper_bounds      5.8     2.9
    lower_bounds      0.5    -2.9
    descriptors       'x1'    'x2'
interface,
  id_interface = 'I1'
  direct
    analysis_driver=  'text_book'
responses,
  id_responses = 'R1'
  num_objective_functions = 1
  no_gradients
  no_hessians
responses,
  id_responses = 'R2'
  num_objective_functions = 1
  analytic_gradients
  analytic_hessians
```



Multi-Objective Optimization

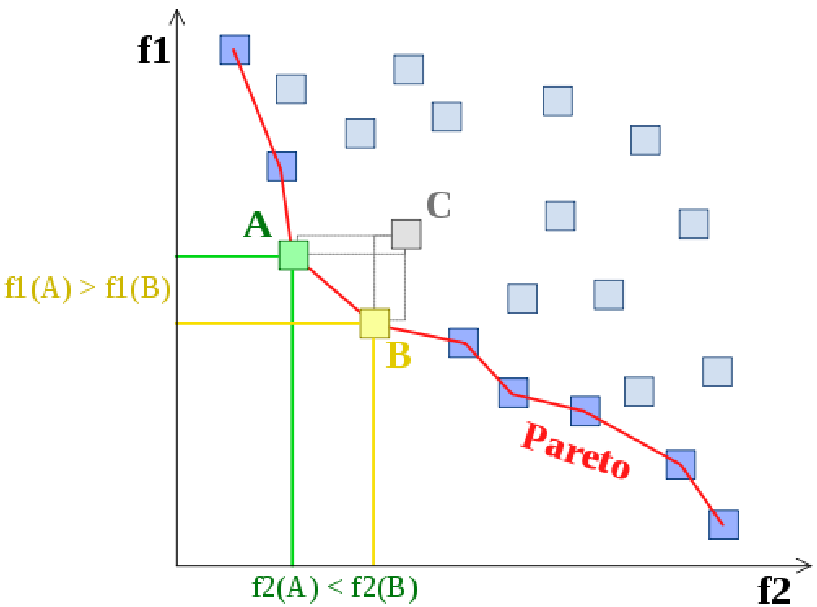


Image from http://en.wikipedia.org/wiki/Pareto_efficiency

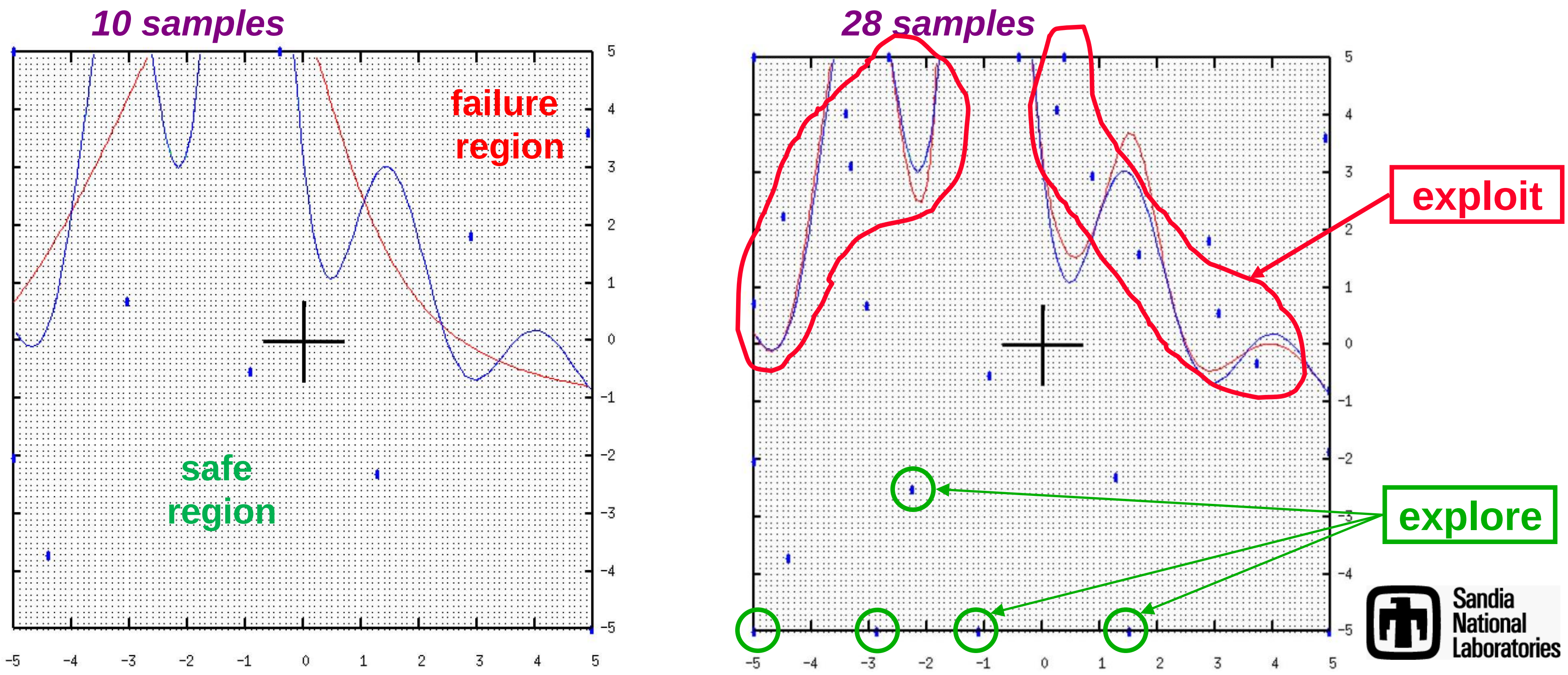
May want tradeoffs between multiple objectives.

```
strategy,
  single_method
  tabular_graphics_data
method,
  optpp_q_newton
    output verbose
    convergence_tolerance = 1.e-8
variables,
  continuous_design = 2
    initial_point    0.9    1.1
    upper_bounds     5.8    2.9
    lower_bounds     0.5   -2.9
    descriptors      'x1'   'x2'
interface,
  system asynchronous
    analysis_driver=  'text_book'
responses,
  num_objective_functions = 3
  multi_objective_weights = .7 .2 .1
  analytic_gradients
  no_hessians
```

Efficient Global Reliability Analysis: GP Surrogate + MMAIS (B.J. Bichon)

- Apply an EGO-like method to the equality-constrained optimization problem
- In EGRA, an expected feasibility function balances exploration with local search near the failure boundary to refine the GP
- Cost competitive with best MPP search methods, yet better probability of failure estimates; addresses nonlinear and multimodal challenges

Gaussian process model (level curves) of reliability limit state with



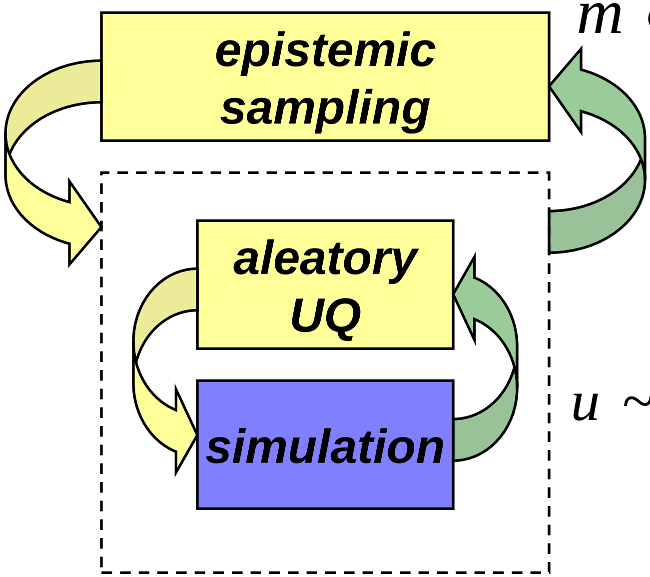
Sandia National Laboratories

Epistemic UQ: Nested ("Second-order")Approaches

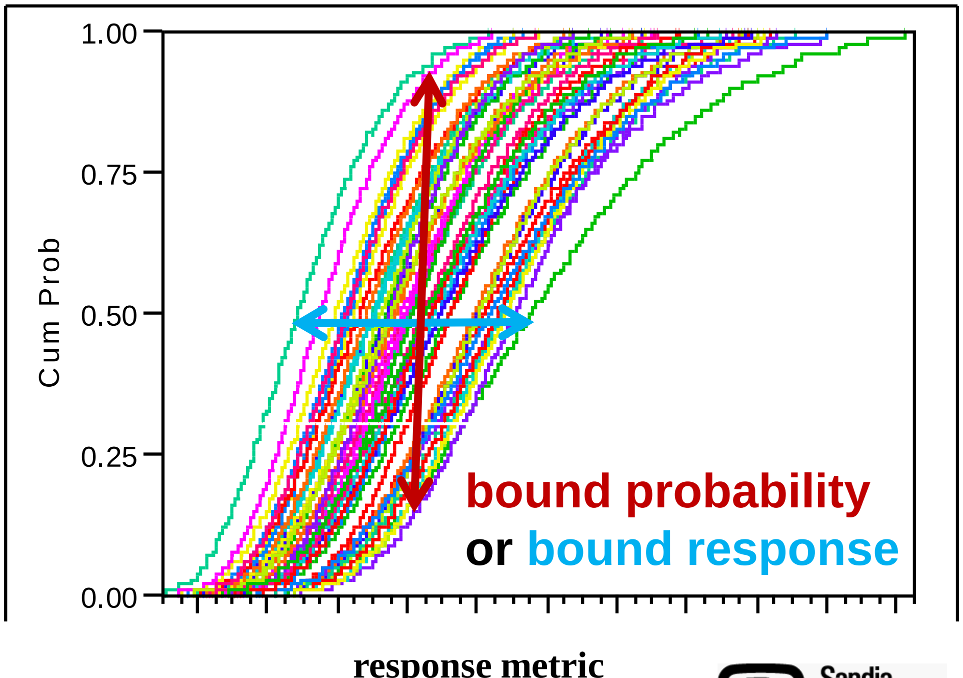
- Propagate over epistemic and aleatory uncertainty, e.g.,
  UQ with bounds on the mean of a normal distribution (hyper-parameters)
- Typical in regulatory analyses (e.g., NRC. WIPP)
- Outer loop: epistemic (interval) variables, inner loop UQ over aleatory (probability) variables; *potentially costly, not conservative*
- *If treating epistemic as uniform, do not analyze probabilistically!*

50 outer loop samples:
50 aleatory CDF traces

$m \in [L, U]$



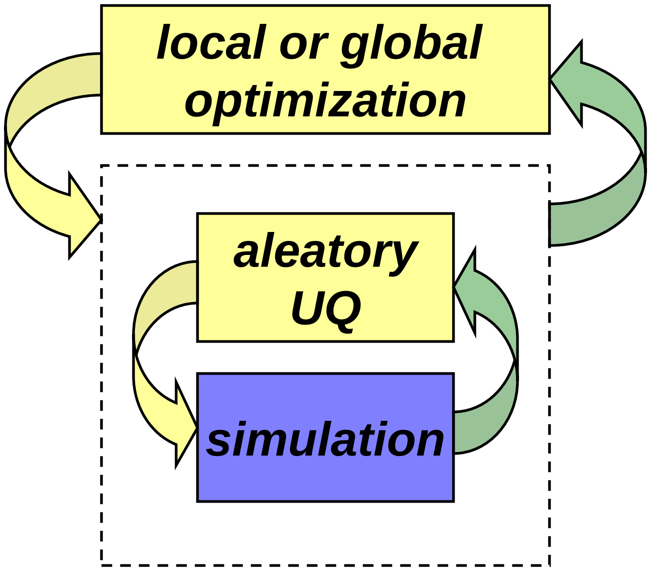
$u \sim N(m, \sigma)$



"Envelope" of CDF traces represents response epistemic uncertainty

Sandia National Laboratories

Interval Estimation Approach (Probability Bounds Analysis)



- ***Propagate intervals through simulation code***
- **Outer loop: determine interval on statistics, e.g., mean, variance**
  - **global optimization problem: find max/min of statistic of interest, given bound constrained interval variables**
  - **use EGO to solve 2 optimization problems with essentially one Gaussian process surrogate**
- **Inner loop: Use sampling, PCE, etc., to determine the CDFs or moments with respect to the aleatory variables**

$$\min_{u_E} \; f_{STAT}(u_A \mid u_E)$$
$$u_{LB} \le u_E \le u_{UB}$$
$$u_A \sim F(u_A; u_E)$$

$$\max_{u_E} \; f_{STAT}(u_A \mid u_E)$$
$$u_{LB} \le u_E \le u_{UB}$$
$$u_A \sim F(u_A; u_E)$$

Sandia National Laboratories

Many Types of Data-Fit Surrogates

Polynomials are accurate in small regions and smooth noisy data.

linear

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i$$

quadratic

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{i=1}^{n} \sum_{j \geq i}^{n} c_{ij} x_i x_j$$

cubic

$$\hat{f}(\mathbf{x}) \approx c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{i=1}^{n} \sum_{j \geq i}^{n} c_{ij} x_i x_j + \sum_{i=1}^{n} \sum_{j \geq i}^{n} \sum_{k \geq j}^{n} c_{ijk} x_i x_j x_k$$

Splines can represent complex multi-modal surfaces and smooth noisy data.

$$\hat{f}(\mathbf{x}) = \sum_{m=1}^{M} a_m B_m(\mathbf{x})$$

truncated power basis functions

Gaussian processes are good predictors of mean and variance but can suffer from ill conditioning.

$$\hat{f}(\underline{x}) \approx \underline{g}(\underline{x})^T \underline{\beta} + \underline{r}(\underline{x})^T \underline{\underline{R}}^{-1} (\underline{f} - \underline{\underline{G}}\, \underline{\beta})$$

trend **correlation**

Correction terms can be applied to surrogates for improved accuracy.

additive

$$\hat{f_{hi_\alpha}}(\mathbf{x}) = f_{lo}(\mathbf{x}) + \alpha(\mathbf{x})$$

multiplicative

$$\hat{f_{hi_\beta}}(\mathbf{x}) = f_{lo}(\mathbf{x}) \beta(\mathbf{x})$$

convex combination

$$\hat{f_{hi_\gamma}}(\mathbf{x}) = \gamma \hat{f_{hi_\alpha}}(\mathbf{x}) + (1-\gamma) \hat{f_{hi_\beta}}(\mathbf{x})$$

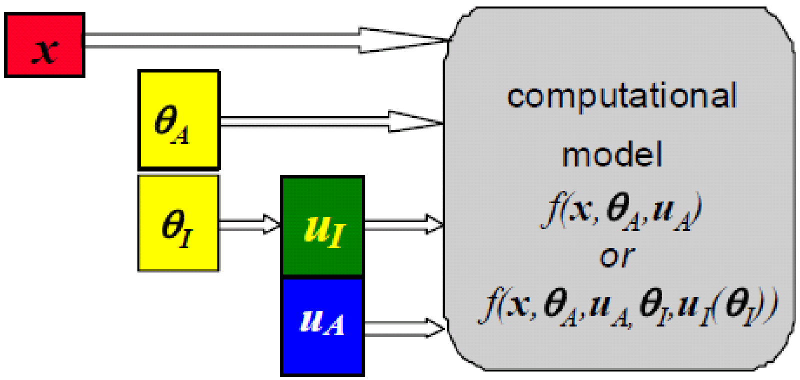
Sandia National Laboratories

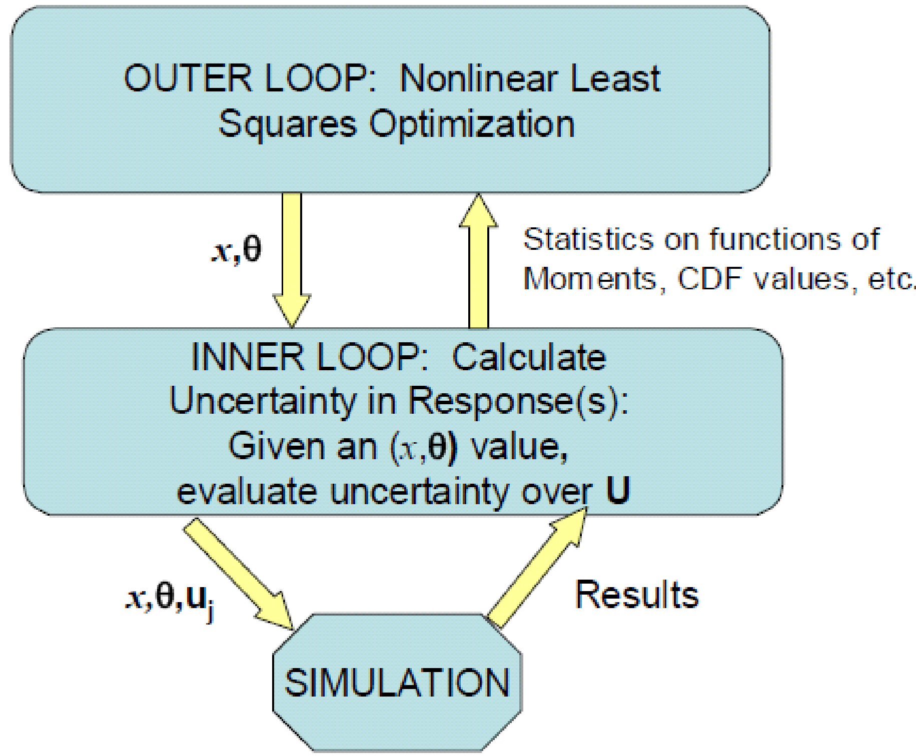
Parameter Estimation Topics

What are the challenges you face in calibration?

- **Dealing with multiple data sets**
- **Data processing and interpolation**
- **Relevant metrics, statistics for comparing**
- **Calibration vs. validation**
- **Tools that would help your process**

Calibration Under Uncertainty





Goal is now to match statistical moments of model over uncertain parameters with statistical moments of target.

Requires a nested solution approach.

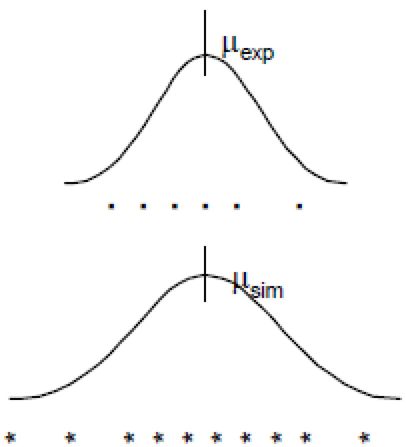
Various Calibration Under Uncertainty Problems

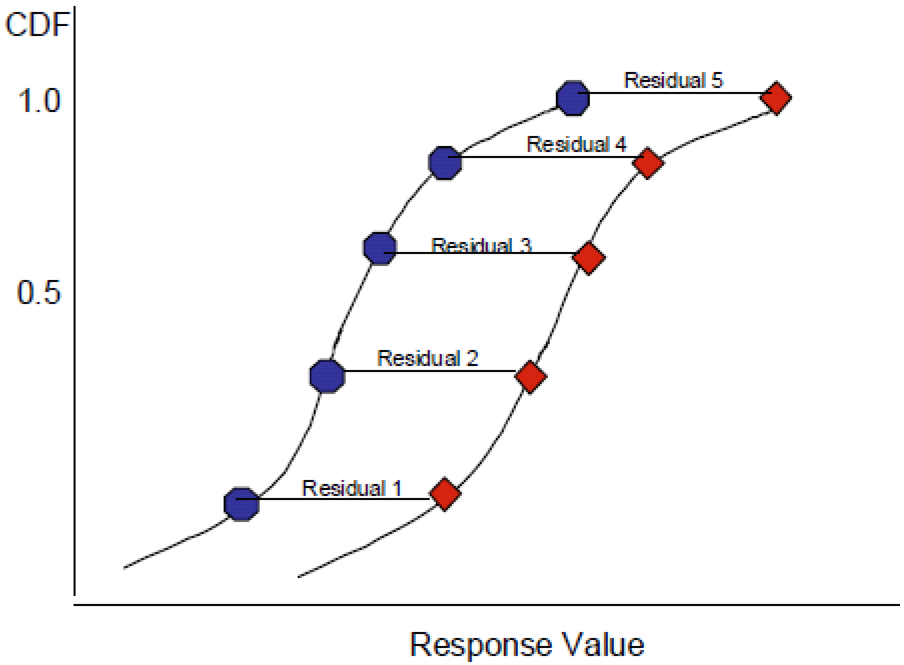
Matching Means

$$S(\theta) = (\mu_{\exp} - \mu_{sim})^2 = \left(\frac{\sum_{i=1}^{n} y_i}{n} - \frac{\sum_{j=1}^{m} f(\mathbf{x};\theta;u_j)}{m} \right)^2$$

Matching Means and Variances

$$S(\theta) = (\mu_{\exp} - \mu_{sim})^2 + (\sigma_{\exp} - \sigma_{sim})^2 .$$





Matching Distributions

$$S(\theta) = \sum_{k=1}^{K} (CDF_{\exp(k)} - CDF_{sim(k)})^2 = \sum_{k=1}^{K} \left([Y_k : \Pr(y \le Y_k) = p_k] - [f_k : \Pr(f(\mathbf{x};\theta;U) \le f_k) = p_k] \right)^2$$

$$S(\theta) = \sum_{k=1}^{K} (CDF_{\exp(k)} - CDF_{sim(k)})^2 = \sum_{k=1}^{K} \left(\Pr(y \le Y_k) - \Pr(f(\mathbf{x};\theta;U) \le Y_k) \right)^2$$

Possible Advanced Topics (dictated by class interest)

General features

- Restart
- Evaluation cache
- Utilities in dakota_restart_util
- Tabular graphics data
- Failure capturing: abort, retry, recover, ignore
- Constraint specification: linear, nonlinear; equality, inequality
- Input/output scaling
- Matlab interface

Approximation methods

- Global data fit surrogate methods (polynomials, MARS, Kriging, etc.)
- Local surrogate methods (Taylor series, multipoint)
- Hierarchical: high/low fidelity models
- Corrections

Strategies/Advanced approaches

- Nested models: OUU
- Multi-objective (Pareto) optimization
- Multistart; multi-level hybrid
- Surrogate-based optimization (variety of constraint handling approaches): trust region; EGO/EGRA
- Reliability-based design optimization
- Advanced UQ topics: polynomial chaos, second-order probability, Dempster-Shafer, surrogate-based UQ
- AMPL: for analytic problems / algebraic mappings

Parallel capabilities: message passing, asynchronous local, hybrid

- Asynchronous evaluations
- Dakota parallel, application serial
- Dakota serial, application parallel
- Multi-level parallel: concurrent iteration, concurrent function evaluations, concurrent analyses,
- multiprocessor simulations

Getting Started and Getting Help

- **Access a Sandia installation: `module avail dakota`**
  AMECH (CA), CEE (ESHPC/SCICO, NM), Computer clusters (both)
  or download

- **Supported on Linux/Unix, Mac OS X,**
  Windows (no MinGW or Cygwin install required)

- **Tour DAKOTA web pages: http://dakota.sandia.gov**
  - **Extensive documentation (user, reference, developer)**
  - **Support mailing lists / archives**
  - **Software downloads: official releases and nightly stable**
    (freely available worldwide via GNU GPL)

- **User's Manual, Chapter 2: Tutorial with example input files**

- **Support:**
  - **dakota-users@software.sandia.gov**
    (DAKOTA team and internal/external user community)
  - **dakota-help@sandia.gov**
    (for SNL-specific or issues involving proprietary information)

Sandia National Laboratories

Course Learning Goals: Did we meet them?

- **Understand tools available in DAKOTA and the kind of design and analysis processes they can support**
- **Requirements for getting started**
- **See the mechanics of running DAKOTA**
- **Where to get help using DAKOTA**

- ***What pieces are still missing or unclear?***